

-2-

Serial No. 10/053,113
Docket No. SVL92001009US1
Firm No. 0055.0050

This listing of claims will replace all prior versions, and listings, of claims in the application:

Listing of Claims:

1. (Previously Presented) A method for querying instances of asset classes in a digital library coupled to a graphical user interface, comprising:

providing at least one asset class having at least one attribute, wherein each attribute is defined to have an attribute object comprising an external data object and attribute object type identifying a type of the attribute object, wherein the attribute object type indicates one of a plurality of different data structure formats searchable through separate application programs, wherein the data structure formats include a multimedia file, a database object accessed through a database application program, and a text object accessed through a text search engine application program;

providing an asset object for each instance of one asset class and information on a file location of attribute objects providing the attributes for the asset object instance, wherein the asset object includes information for the attributes in the instance of the asset class;

presenting the graphical user interface including fields for receiving user input from a window having predefined scrollable items for generating a query indicating an asset name, search predicate, at least one attribute operator, and attribute value, wherein the attribute operator is associated with at least one attribute included in the asset object having the indicated asset name; and

processing the query by:

determining asset object instances of the indicated asset name having one attribute object for the attribute associated with the attribute operator;

accessing the attribute object for the attribute associated with the attribute operator at the file location indicated in the determined asset object instance;

-3-

Serial No. 10/053,113
Docket No. SVL92001009US1
Firm No. 0055.0050

determining from the object type for the attribute associated with the attribute operator one application program for accessing the data structure format of the attribute object type; and calling the determined application program to determine whether the accessed attribute object satisfies the search predicate and the attribute value in the query.

2. (Original) The method of claim 1, wherein the query includes multiple attribute operators and attribute values to query asset object instances whose attribute objects match the attribute values and search predicate for each attribute operator.

3-6. (Canceled)

7. (Previously Presented)) The method of claim 1, wherein one attribute comprises a relationship attribute that defines an association of a first and second asset types and a relationship attribute object associating instances of the first and second asset types, wherein the attribute operator for a relationship attribute is associated with one attribute from the first and second asset types, wherein processing the query to search the attribute object for each asset object instance of the first asset type further comprises:

accessing the relationship attribute object to determine all asset object instances of the second asset type associated with the asset object instance; and

for each determined asset object instance, processing the query by determining the determined asset object instances of the second asset type whose attribute object for the attribute of the second asset type associated with the attribute operator matches the attribute value and satisfies the search predicate.

8. (Original) The method of claim 7, wherein the relationship attribute object comprises a database table, wherein a first column in the database table is for unique identifiers of instances of the first asset type and a second column in the database table is for unique identifiers of instances of the second asset type, wherein a row in the database table identifies one instance of

-4-

Serial No. 10/053,113
Docket No. SVL920010095US1
Firm No. 0055.0050

the first asset type identified by the unique identifier in the first column of the row that is associated with one instance of the second asset type identified by the unique identifier in the second column of the row.

9. (Original) The method of claim 1, wherein the definition of each attribute for an asset class is implemented in an Extensible Markup Language (XML) document, wherein each defined attribute for an asset class comprises a tagged element in the XML document and wherein information for each attribute is embedded in at least one tagged attribute of the tagged element for the attribute.

10. (Canceled)

11. (Previously Presented) The method of claim 1, further comprising:
in response to receiving a user entered asset name for the query, determining attribute operators associated with attributes of the asset name;
generating a display of a list of the determined attribute operators; and
receiving user selection of one of the determined attribute operators from the displayed list, wherein the user selected attribute operator is used in the query.

12. (Previously Presented) A system for querying data, comprising:
a processor;
a digital library including asset classes;
a graphical user interface coupled to the digital library;
a definition of at least one asset class having at least one attribute, wherein each attribute is defined to have an attribute object comprising an external data object and attribute object type identifying a type of the attribute object, wherein the attribute object type indicates one of a plurality of different data structure formats searchable through separate application programs, wherein the data structure formats include a multimedia file, a database object accessed through a

database application program, and a text object accessed through a text search engine application program;

an asset object for each instance of one asset class and information on a file location of attribute objects providing the attributes for the asset object instance, wherein the asset object includes information for the attributes in the instance of the asset class;

computer implemented code executed by the processor to perform:

(i) presenting the graphical user interface including fields for receiving user input from a window having predefined scrollable items for generating a query indicating an asset name, search predicate, at least one attribute operator, and attribute value, wherein the attribute operator is associated with at least one attribute included in the asset object having the indicated asset name; and

(ii) processing the query by:

determining asset object instances of the indicated asset name having one attribute object for the attribute associated with the attribute operator;

accessing the attribute object for the attribute associated with the attribute operator at the file location indicated in the determined asset object instance;

determining from the object type for the attribute associated with the attribute operator one application program for accessing the data structure format of the attribute object type; and

calling the determined application program to determine whether the accessed attribute object satisfies the search predicate and the attribute value in the query.

13. (Canceled)

14. (Original) The system of claim 12, wherein the query includes multiple attribute operators and attribute values to query asset object instances whose attribute objects match the attribute values and search predicate for each attribute operator.

15-18. (Canceled)

19. (Previously Presented) The system of claim 12, wherein one attribute comprises a relationship attribute that defines an association of a first and second asset types and a relationship attribute object associating instances of the first and second asset types, wherein the attribute operator for a relationship attribute is associated with one attribute from the first and second asset types, wherein the means for processing the query to search the attribute object for each asset object instance of the first asset type performs:

accessing the relationship attribute object to determine all asset object instances of the second asset type associated with the asset object instance; and

for each determined asset object instance, processing the query by determining the determined asset object instances of the second asset type whose attribute object for the attribute of the second asset type associated with the attribute operator matches the attribute value and satisfies the search predicate.

20. (Original) The system of claim 19, wherein the relationship attribute object comprises a database table, wherein a first column in the database table is for unique identifiers of instances of the first asset type and a second column in the database table is for unique identifiers of instances of the second asset type, wherein a row in the database table identifies one instance of the first asset type identified by the unique identifier in the first column of the row that is associated with one instance of the second asset type identified by the unique identifier in the second column of the row.

21. (Original) The system of claim 12, wherein the definition of each attribute for an asset class is implemented in an Extensible Markup Language (XML) document, wherein each defined attribute for an asset class comprises a tagged element in the XML document and wherein information for each attribute is embedded in at least one tagged attribute of the tagged element for the attribute.

22. (Canceled)

23. (Previously Presented) The system of claim 12, further comprising:
means for determining attribute operators associated with attributes of the asset name in response to receiving a user entered asset name for the query;
means for generating a display of a list of the determined attribute operators; and
means for receiving user selection of one of the determined attribute operators from the displayed list, wherein the user selected attribute operator is used in the query.

24. (Currently Amended) A computer readable storage medium including code that is executed to query instances of asset classes in a digital library coupled to a graphical user interface, wherein the code causes operations comprising:

providing at least one asset class having at least one attribute, wherein each attribute is defined to have an attribute object comprising an external data object and attribute object type identifying a type of the attribute object, wherein the attribute object type indicates one of a plurality of different data structure formats searchable through separate application programs, wherein the data structure formats include a multimedia file, a database object accessed through a database application program, and a text object accessed through a text search engine application program;

providing an asset object for each instance of one asset class and information on a file location of attribute objects providing the attributes for the asset object instance, wherein the asset object includes information for the attributes in the instance of the asset class;

presenting the graphical user interface including fields for receiving user input from a window having predefined scrollable items for generating a query indicating an asset name, search predicate, at least one attribute operator, and attribute value, wherein the attribute operator is associated with at least one attribute included in the asset object having the indicated asset name; and

processing the query by:

determining asset object instances of the indicated asset name having one attribute object for the attribute associated with the attribute operator;

accessing the attribute object for the attribute associated with the attribute operator at the file location indicated in the determined asset object instance;

determining from the object type for the attribute associated with the attribute operator one application program for accessing the data structure format of the attribute object type; and

calling the determined application program to determine whether the accessed attribute object satisfies the search predicate and the attribute value in the query.

25. (Previously Presented) The computer readable storage medium of 24, wherein the query includes multiple attribute operators and attribute values to query asset object instances whose attribute objects match the attribute values and search predicate for each attribute operator.

26-29. (Canceled)

30. (Previously Presented) The computer readable storage medium 24, wherein one attribute comprises a relationship attribute that defines an association of a first and second asset types and a relationship attribute object associating instances of the first and second asset types, wherein the attribute operator for a relationship attribute is associated with one attribute from the first and second asset types, wherein processing the query to search the attribute object for each asset object instance of the first asset type further comprises:

accessing the relationship attribute object to determine all asset object instances of the second asset type associated with the asset object instance; and

for each determined asset object instance, processing the query by determining the determined asset object instances of the second asset type whose attribute object for the attribute

-9-

Serial No. 10/053,113
Docket No. SVL920010095US1
Firm No. 0055.0050

of the second asset type associated with the attribute operator matches the attribute value and satisfies the search predicate.

31. (Previously Presented) The computer readable storage medium 30, wherein the relationship attribute object comprises a database table, wherein a first column in the database table is for unique identifiers of instances of the first asset type and a second column in the database table is for unique identifiers of instances of the second asset type, wherein a row in the database table identifies one instance of the first asset type identified by the unique identifier in the first column of the row that is associated with one instance of the second asset type identified by the unique identifier in the second column of the row.

32. (Previously Presented) The computer readable storage medium 24, wherein the definition of each attribute for an asset class is implemented in an Extensible Markup Language (XML) document, wherein each defined attribute for an asset class comprises a tagged element in the XML document and wherein information for each attribute is embedded in at least one tagged attribute of the tagged element for the attribute.

33. (Canceled)

34. (Previously Presented) The computer readable storage medium 33, further comprising:

in response to receiving a user entered asset name for the query, determining attribute operators associated with attributes of the asset name;

generating a display of a list of the determined attribute operators; and

receiving user selection of one of the determined attribute operators from the displayed list, wherein the user selected attribute operator is used in the query.